JUN 16 2010
PATENT & TRADEMARK OFFICE

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of	Conf. No.: 1484
BONSMA, et al.	Atty. Ref.: LSN-36-1993
Serial No. 10/583,964	TC/A.U.: 2442
Filed: June 21, 2006	Examiner: Jeffrey L. Nickerson

For: DISTRIBUTED COMPUTER SYSTEM FOR STORING DATA ITEMS ASSIGNED TO VIRTUAL DIRECTORIES

* *

June 16, 2010

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF UNDER 37 C.F.R. § 41.37(c)

Sir:

Applicant has appealed to the Board of Patent Appeals and Interferences (Notice of Appeal filed April 16, 2010) from the last decision of the Examiner (Final Office Action dated February 16, 2010 and Advisory Action dated April 13, 2010). An appeal brief pursuant to 37 C.F.R. § 41.37(c) is now presented.

06/17/2010 LNGUYEN1 00000073 10583964
01 FC:1402 540.00 OP

TABLE OF CONTENTS

(I) REAL PARTY IN INTEREST

The real party in interest is British Telecommunications public limited company, a British corporation of the United Kingdom.

(II) RELATED APPEALS AND INTERFERENCES

The appellant, the undersigned, and the assignee are not aware of any related appeals, interferences, or judicial proceedings (past or present), which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

(III) STATUS OF CLAIMS

Claims 1-21 are pending and have been rejected. No claims have been substantively allowed. The rejections of claims 1-21 are being appealed.

(IV) STATUS OF AMENDMENTS

An Amendment under 37 C.F.R. § 1.116 was filed on March 15, 2010 -- i.e., after the date of the Final Rejection. The Advisory Action dated April 13, 2010 indicates that the amendments proffered therein would be entered for purposes of appeal (see Box 7(b)) and overcame the rejection under 35 U.S.C. § 112, second paragraph. Thus, the status of the amendments is that the March 15, 2010 Amendment was entered, and the claims on appeal herein are those claims presented in Applicant's March 15, 2010 Amendment.

(V) SUMMARY OF CLAIMED SUBJECT MATTER

Each independent claim, each dependent claim argued separately, and each claim having means plus function language is summarized below including exemplary reference(s) to page and line number(s) of the specification.

A. Introduction

Certain exemplary embodiments relate to a distributed computer system that includes computers that store data items, with each such data item being assigned to one of a plurality of virtual directories. Each computer has at least one node of a virtual network for directory look-up. For each node involved in directory look-up, linking data comprising addresses of other nodes associated with the same virtual directory is provided. Software[1] also is provided to such nodes, and such software in response to an enquiry message either replies identifying itself or forwards the message to another node of the network, according to whether the node is or is not associated with the directory specified in the enquiry message.

Similarly, each computer has, for each item that it stores, a node of a virtual network for item look-up. The item look-up is separate from the directory look-up. For each node involved in item look-up, linking data comprising addresses of other such nodes associated with items assigned to the same virtual directory is provided, such that these linking data together define a plurality of virtual networks for item look-up, with

[1] As those in the art will understand, "software" is used in the colloquial sense as a shorthand expression for executable computer program code structures stored on non-transitory memory accessible to at least one CPU for execution of functionality in associated apparatus.

each such network corresponding to a respective different virtual directory. The node also has software that, in response to an enquiry message either replies with the item sought (or identifies itself for later item retrieval) or forwards the message to another node of the network, according to whether the node is or is not associated with the item specified in the enquiry message.

A retrieval mechanism also may be provided for certain exemplary embodiments. Each node matching a given label has associated with it a data storage area for containing the label and is responsive to enquiry messages to return a list of addresses of other nodes matching the same label. First retrieval means responds to input of a label to identify the address of a node matching that label, and second retrieval means then sends an enquiry message to the address identified by the first retrieval means and upon receipt of a response to iteratively send further enquiry messages to the node having the addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message.

B. Independent Claim 1

Independent claim 1 relates to a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computers (e.g., computer in Fig. 1; p. 2, lines 3-13) are provided, with each storing data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), and with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). Each computer that has a said data item stored thereon has at least one

node of a virtual network for directory look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17), with said node for directory look-up comprising: (i) data identifying that one of the plurality of virtual directories with which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by: (a) in response to an enquiry message that identifies another of the virtual directories forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). Each computer that has a said data item stored thereon has, for each item stored thereon, a node of a virtual network for item look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for item look-up comprises (i) data identifying the item with which the node for item look-up is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The

software operates by (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), and (b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the identified item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). At least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21), (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19), and (iii) to receive the reply message containing the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

C. Independent Claim 2

Independent claim 2 relates to a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computers (e.g., computer in Fig. 1; p. 2, lines 3-13), with each storing data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), and with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). Each computer that has a said data item stored thereon having at least one node of a virtual network for directory look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for directory look-up comprises (i) data identifying that one

of a plurality of the virtual directories with which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). Each computer that has a said data item stored thereon has, for each item stored thereon (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17), a node of a virtual network for item look-up. Said node for directory look-up comprises (i) data identifying the item with which the node for directory look-up is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the items forwarding the message to another node for directory look-up of the network (e.g., 1110 and 1120 in

Fig. 11, p. 19, lines 23-29), (b) in response to an enquiry message that identifies the item with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for item look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4), and (c) in response to a request message that identifies the item with which the node for directory look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). At least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21), (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19), and (iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

D. Independent Claim 6

Independent claim 6 relates to a computer for use in a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A data store stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). At least one node of a virtual network is provided for directory look-up (e.g., computer in Fig. 1; p. 2, lines 3-13). Said node for directory look-up comprises (i) data identifying that one of the plurality of virtual directories with

which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). For each item stored on said computer, a node of a virtual network is provided for item look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for item look-up comprises (i) data identifying the item with which the node for item look-up is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), and (b) in response to an enquiry message that identifies the item with which the node for

item look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). Retrieval means are responsive to receipt of a query identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21), (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19), and (iii) to receive the reply message containing the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

E. Independent Claim 7

Independent claim 7 relates to a computer for use in a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A data store stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). At least one node of a virtual network is provided for directory look-up (e.g., computer in Fig. 1; p. 2, lines 3-13). Said node for directory look-up comprises (i) data identifying that one of a plurality of the virtual directories with which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message

to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). For each item stored on said computer, a node of a virtual network is provided for item look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for item look-up comprises (i) data identifying the item with which the for item look-up node is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), (b) in response to an enquiry message that identifies the item with which the node for item look-up is associated generating a reply message identifying a computer that the node for item look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4), and (c) in response to a request message identifying the item with which the node for item look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). Retrieval means are responsive to receipt of a query

identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21); (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19); and (iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

F. Independent Claim 11

Independent claim 11 relates to a distributed computer system (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computing nodes (e.g., computer in Fig. 1; p. 2, lines 3-13) is provided, wherein each computer stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). First retrieval means are responsive to input of a directory name to identify a computing node having items in that directory (e.g., 1902, 1904, and 1906 in Fig. 19; p. 24, lines 5-18). Second retrieval means are connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in the same directory (e.g., 1910 in Fig. 19; p. 34, lines 19-23). Each computing node having items in a given directory has associated with it a data storage area for containing addresses for other computing nodes having items in the same directory and is responsive to enquiry messages to return a message containing the addresses of the list (e.g., 22 and 24 / 25 and

27 in Fig. 1; p. 2, lines 6-13; p. 34, lines 24-25). The second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in the directory in question (e.g., 1912 and 1914 in Fig. 19; p. 34, line 24 to p. 35, line 14).

G. Independent Claim 13

Independent claim 13 relates to a distributed computer system (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computing nodes (e.g., computer in Fig. 1; p. 2, lines 3-13) is provided. Each computer stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17), and with some of said directories being assigned, as subdirectories, to another of said directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). First retrieval means are responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory (e.g., 2002, 2004, and 2006 in Fig. 20; p. 35, lines 15-27). Second retrieval means are connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in at least one subdirectory assigned to the same directory (e.g., 2010 in Fig. 20; p. 35, lines 15-27). Each computing node having items in at least one subdirectory assigned to a given directory has associated with it a data storage area for containing addresses for

other computing nodes having items in at least one subdirectory assigned to the same directory and is responsive to enquiry messages to return a message containing the addresses of the list (e.g., 22 and 24 / 25 and 27 in Fig. 1; p. 2, lines 6-13; p. 35, lines 15-27). The second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in subdirectories of the directory in question (e.g., 2012 and 2014 in Fig. 20; p. 35, lines 15-27).

data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). Each computer that has a said data item stored thereon has, for each item stored thereon (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17), a node of a virtual network for item look-up. Said node for directory look-up comprises (i) data identifying the item with which the node for directory look-up is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the items forwarding the message to another node for directory look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), (b) in response to an enquiry message that identifies the item with which the node for directory look-up is associated, generating a reply message

identifying the computer that the node for item look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4), and (c) in response to a request message that identifies the item with which the node for directory look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). At least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21), (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19), and (iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

D. Independent Claim 6

Independent claim 6 relates to a computer for use in a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A data store stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). At least one node of a virtual network is provided for directory look-up (e.g., computer in Fig. 1; p. 2, lines 3-13). Said node for directory look-up comprises (i) data identifying that one of the plurality of virtual directories with which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for

directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). For each item stored on said computer, a node of a virtual network is provided for item look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for item look-up comprises (i) data identifying the item with which the node for item look-up is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), and (b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). Retrieval means are responsive to receipt of a query

identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in Fig. 18; p. 33, lines 14-21), (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19), and (iii) to receive the reply message containing the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

E. Independent Claim 7

Independent claim 7 relates to a computer for use in a distributed computer system comprising a plurality of computers (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A data store stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). At least one node of a virtual network is provided for directory look-up (e.g., computer in Fig. 1; p. 2, lines 3-13). Said node for directory look-up comprises (i) data identifying that one of a plurality of the virtual directories with which the node for directory look-up is associated (e.g., "foo/bar" directory in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for directory look-up (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates by (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network (e.g., 1802 and 1804 in Fig. 18; p. 32, lines 12-18), and (b) in response to an enquiry message that identifies the virtual

directory with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for directory look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4). For each item stored on said computer, a node of a virtual network is provided for item look-up (e.g., p. 2, lines 3-13; p. 2, lines 23-33; p. 3, line 17 to p. 4, line 17). Said node for item look-up comprises (i) data identifying the item with which the for item look-up node is associated (e.g., "foo/bar:abc" item in Fig. 17; p. 28, line 9 to p. 30, line 27), (ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory (e.g., "links" in Fig. 17; p. 28, line 9 to p. 30, line 27), and (iii) software (e.g., 23 / 25 in Fig. 1; page 2, lines 6-13). The software operates (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network (e.g., 1110 and 1120 in Fig. 11, p. 19, lines 23-29), (b) in response to an enquiry message that identifies the item with which the node for item look-up is associated generating a reply message identifying a computer that the node for item look-up is located on (e.g., 1810 and 1812 in Fig. 18; p. 32, line 23 to p. 33, line 4), and (c) in response to a request message identifying the item with which the node for item look-up is associated, generating a reply message including the item (e.g., 1115 in Fig. 11; p. 19, lines 25-27). Retrieval means are responsive to receipt of a query identifying a directory and an item within that directory to (i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory (e.g., 1820 in

Fig. 18; p. 33, lines 14-21); (ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item (e.g., p. 33, lines 14-19); and (iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item (e.g., 1822 in Fig. 18; p. 33, lines 20-21).

F. Independent Claim 11

Independent claim 11 relates to a distributed computer system (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computing nodes (e.g., computer in Fig. 1; p. 2, lines 3-13) is provided, wherein each computer stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). First retrieval means are responsive to input of a directory name to identify a computing node having items in that directory (e.g., 1902, 1904, and 1906 in Fig. 19; p. 24, lines 5-18). Second retrieval means are connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in the same directory (e.g., 1910 in Fig. 19; p. 34, lines 19-23). Each computing node having items in a given directory has associated with it a data storage area for containing addresses for other computing nodes having items in the same directory and is responsive to enquiry messages to return a message containing the addresses of the list (e.g., 22 and 24 / 25 and 27 in Fig. 1; p. 2, lines 6-13; p. 34, lines 24-25). The second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon

receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in the directory in question (e.g., 1912 and 1914 in Fig. 19; p. 34, line 24 to p. 35, line 14).

G. Independent Claim 13

Independent claim 13 relates to a distributed computer system (e.g., network 10 in Fig. 1; p. 1, line 16 to p. 2, line 13). A plurality of computing nodes (e.g., computer in Fig. 1; p. 2, lines 3-13) is provided. Each computer stores data items (e.g., 20 in Fig. 1; p. 2, lines 3-13), with each data item being assigned to one of a plurality of virtual directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17), and with some of said directories being assigned, as subdirectories, to another of said directories (e.g., p. 2, lines 23-33 and p. 3, line 17 to p. 4, line 17). First retrieval means are responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory (e.g., 2002, 2004, and 2006 in Fig. 20; p. 35, lines 15-27). Second retrieval means are connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in at least one subdirectory assigned to the same directory (e.g., 2010 in Fig. 20; p. 35, lines 15-27). Each computing node having items in at least one subdirectory assigned to a given directory has associated with it a data storage area for containing addresses for other computing nodes having items in at least one subdirectory assigned to the same directory and is responsive to enquiry messages to return a message containing the

addresses of the list (e.g., 22 and 24 / 25 and 27 in Fig. 1; p. 2, lines 6-13; p. 35, lines 15-27). The second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in subdirectories of the directory in question (e.g., 2012 and 2014 in Fig. 20; p. 35, lines 15-27).

(VI) GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

First, claims 1-4, 6-8, 11-12, and 16-18 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma et al. ("A distributed implementation of the SWAN peer-to-peer look-up system using mobile agents"), Triantafillou et al. ("Towards high performance peer-to-peer content and resource sharing systems"), Kwon et al. ("An efficient peer-to-peer file sharing exploiting hierarchy and asymmetry"), and Adar et al. ("Free Riding on Gnutella").

Second, claims 5, 9-10, and 13-15 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, and Christenson et al. (U.S. Publication No. 2002/0112008).[1]

Third, claims 19-20 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, and Bonsma '669 (WO 03/034669).

Fourth, claim 21 stands rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, Bonsma '669, and Yemini et al. (U.S. Publication No. 2002/0163889).[2]

---

[1] The Advisory Action makes clear that the fifth reference being relied upon in this five-way § 103 rejection is Christenson and that Official Notice is not being relied upon.

[2] As indicated above, Applicant's March 15, 2010 Amendment was entered for purposes of appeal and overcame the previous rejection under 35 U.S.C. § 112, second paragraph.

(VII) ARGUMENT

A. Claims 1-4, 6-8, 11-12, and 16-18 Each Are Not "Obvious" Over Bonsma, Triantafillou, Kwon, and Adar.

Claims 1-4, 6-8, 11-12, and 16-18 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma et al. ("A distributed implementation of the SWAN peer-to-peer look-up system using mobile agents"), Triantafillou et al. ("Towards high performance peer-to-peer content and resource sharing systems"), Kwon et al. ("An efficient peer-to-peer file sharing exploiting hierarchy and asymmetry"), and Adar et al. ("Free Riding on Gnutella"). This four-way § 103 rejection is erroneous and should be reversed for at least the following reasons.

The Final Office Action correctly notes that Bonsma -- the primary reference relied upon in this four-way § 103 rejection -- describes the SWAN distributed computer network devised by the instant inventors/Applicant. The Final Office Action also correctly concedes that Bonsma fails to teach or suggest one and one-half pages worth of features expressly called for in claim 1. See pages 7-8 of the Final Office Action.

To make up for the numerous "holes" in Bonsma, the Final Office Action adopts the approach of decomposing whole claims into separated limitations, further decomposing those separated limitations into even more fragmented bits and pieces, and then searching through the prior art for teachings or suggestions using similar words. Aside from violating the fundamental statutory requirement that the obviousness of a claim is to be determined based on the claim "as a whole," this improper hindsight approach to reconstructing the claims from disconnected and fragmented bits and pieces of the claim language itself has resulted in the Examiner's reliance on inconsistent and

self-contradictory teachings of the cited references. For example, as shown in much more detail below, the Final Office Action maps multiple separate claim elements to a single element in the cited art. As another example, and again as shown in much more detail below, the Final Office Action maps various sub-steps of fundamentally non-decomposable processes in the cited references to various parts of claim requirements.

The Examiner justifies this ex post approach to "examination" by alleging in the Advisory Action that these prior art teachings constitute "added functionality" that (presumably) would have been combined by one of ordinary skill in the art at the time of the invention. But even a cursory reading of the relied-upon references reveals that the prior art teachings typically are part of a fundamental process that cannot be artificially decomposed and that certainly could not (and would not) have been extracted and arbitrarily recombined in the manner alleged in the Final Office Action.

Perhaps recognizing these serious problems with the way that the claims have been treated, the Advisory Action now advances the theory that whole portions of the claims can be ignored simply because Applicant has claimed software and/or other means "operable" to perform specific tasks. In particular, the Advisory Action indicates that "any programmable device is operable to perform the function if programmed correctly." This assertion ignores the fact that software "operable" to do something necessarily is "programmed" or "programmatically configured" to do that particular something and cannot be "operable" to do something else unless it is "re-programmed" or "programmatically re-configured" in a wholly different way. For instance, the software being used to draft this Appeal Brief certainly is not "operable" to administer the system

of claim 1 (or any of the nodes therein), and vice versa. This specious argument should not distract from the improper examination of the pending claims and the numerous deficiencies of the cited art, alone and in combination.

For example, claim 1 relates to a distributed computer system comprising a plurality of computers in which, *inter alia*, "each computer that has a said data item stored thereon [has] at least one node of a virtual network for directory look-up, . . . [and] each computer that has a said data item stored thereon [has], for each item stored thereon, a node of a virtual network for item look-up. . . ." These features of claim 1 are not present in the cited art, alone or when taken in combination. Thus, even the allegedly "obvious" four-way combination fails to render obvious claim 1.

Bonsma discloses that a collection of virtual nodes is connected by a communication infrastructure, and that each node is given a potentially arbitrary location-independent node identity. Bonsma discloses a virtual network created from long- and short-ranged links between nodes. The nodes are organized based on their identities. But Bonsma does not teach or suggest the specific hierarchy of the nodes for directory look-up (and the nodes for item look-up) required by the present claims. *Compare* Bonsma at sections 3 and 3.1, *with* at least the above-quoted language of claim 1. These cited portions of Bonsma fail to teach or suggest a computer storing data items, the computer having, for each item stored thereon, a node of a virtual network for item look-up and the computer having, for each virtual directory the stored items are assigned to, a node of a virtual network for item look-up.

The Final Office Action asserts that the teachings of Triantafillou and Kwon when combined with Bonsma disclose the "node for directory look-up" and "node for item look-up" limitations of claim 1, and the specific components that these nodes comprise. However, Triantafillou merely provides for the processing of queries identifying documents, with those documents belonging to document categories. Triantafillou does not provide for the processing of queries specifically identifying document categories. Thus, even assuming, *arguendo*, that the Triantafillou/Kwon system taught one type of node that corresponded to a certain element of claim 1, such could only correspond to one of the two types of node required in claim 1 -- i.e., the "node for item look-up" element of claim 1 -- and simply could not at the same time also correspond to the different "node for directory look-up" element of claim 1.

The Final Office Action also asserts that section 3.3 of Triantafillou teaches the structure of the "node for directory look-up" specifically called for in claim 1, which requires "(i) data identifying that one of the plurality of virtual directories with which the node for directory look-up is associated, (ii) linking data comprising addresses of other such nodes for directory look-up, and (iii) software operable (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network, and (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on. . . ."

This assertion is incorrect. Section 3.3 of Triantafillou discloses a two-step process of query processing. The first step is performed by a requesting node that sends a query, and the second step is performed by a target node that receives said query. Neither the requesting node nor the target node of Triantafillou corresponds to the node for directory look-up called for in claim 1.

More particularly, Triantafillou discloses as step 1 of its two-step procedure:

1. The requesting node does the following:
   a. It maps the keywords to one or more semantic categories using appropriate categorization tool (...) . . .
   b. Through its DCRT (Document Category Routing Table) it finds clusters of nodes with the semantic categories.
   c. It chooses a random number from each associated cluster, using its NRT (Number Routing Table) and sends the query to it. If no live node exists, the query will fail. The random selection of nodes can ensure that cluster nodes get an equal share of the workload targeting their cluster.

Thus, Triantafillou at Section 3.3 does not teach or suggest computer program code that, "in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generat[es] a reply message identifying a computer that the node for directory look-up is located on. . ." as required in claim 1.

Triantafillou further discloses as step 2:

2. The target node does the following:
   a. It matches the categories of the query against the semantic categories of its documents and finds a number a of resulting documents matching the query.
   b. If the number of resulting documents is less than m, (the target node) forwards the query to all of its known neighbors in the cluster, decreasing m by a. This will be recursively repeated until the desired number of documents is found or all reachable nodes of the cluster have been queried (...) . . .
   c. The final result set is returned to the requesting node by (the target) node.

Therefore, the target node of Triantafillou returns only the final result set comprising documents specified in the query. That is, the target node only returns a reply comprising entities corresponding to the "items" of claim 1. Thus, the target node of Triantafillou simply does not include software operable to generate a reply message identifying the computer the node for directory look-up is located on, as required by claim 1. Moreover, the target node of Triantafillou does not include software operable, in response to an enquiry message that identifies another of the virtual directories, to forward the message to another node for directory look-up of the network. Indeed, Triantafillou teaches that the requesting node only forwards the query to its neighbors if the number of resulting documents is less than a specific number.

Although the Final Office Action notes that a node may be a logical entity, this observation misses the point. Irrespective of how the functions described in Triantafillou are distributed among nodes, Triantafillou does not describe all of the functions provided by the distributed computer system of claim 1.

That is partly because Triantafillou describes a single set of clusters of nodes with a single function, namely, to serve retrieval requests -- i.e., to respond to a request by supplying the requested document, as noted in the first paragraph of section see 3.1. This teaching actually corresponds most closely to the later recited node in claim 1 of the (second) virtual network, i.e., the virtual network for item look-up that involves:

> (a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network; [and]

> (b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the identified item.

As defined in claim 1 in part (ii) immediately preceding the above-quoted part, the "network" referred to here (i.e., the second virtual network) is a virtual network for item look-up corresponding to a single virtual directory. This second virtual network may be compared to a cluster of nodes that functions solely to respond to a request by supplying the requested document (although the second virtual network differs from the clusters of Triantafillou in other ways).

The clusters of nodes do not correspond to the earlier recited (primary) virtual network for virtual directory look-up. In particular, nowhere in Triantafillou is there an indication that the clusters of nodes act to provide the following features of claim 1:

> (a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network; [and]
>
> (b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for directory look-up is located on..

Translating such requirements into the language of Triantafillou would result in something like a cluster of nodes in which a node of the cluster responds to a request identifying a cluster with which it is associated (i.e., by use of a cluster-id) by returning the identity of the computer at which the node is located, and in which the node responds to a request identifying a different cluster by forwarding the request to another node.

No such capability is described in Triantafillou, where a cluster only redirects requests that identify items (not clusters) and only to other cluster nodes -- i.e., nodes of the same cluster.

Triantafillou's cluster nodes provide the requested document (if the document is stored on the node) or find another node within the same cluster that can supply it. This is achieved by storing cluster metadata describing which documents are stored by which cluster nodes. The metadata is described at section 3.2.

The Examiner relies on the metadata description of section 3.2 and the query processing of section 3.3, but uses the same sections of Triantafillou to represent different features of claim 1. The various interpretations advanced during the prosecution of this application are listed below. These sections of Triantafillou and the ideas discussed therein do not correspond to the features of claim 1 when that claim is read as a whole and in context. Quite the contrary, as will be appreciated from the discussion below, these interpretations actually are in part mutually exclusive or mutually contradictory. Despite Applicant's requests, the Examiner still has not clarified which claim features allegedly correspond with which sections of Triantafillou.

From section 3.3 Target Node:

The Final Office Action identifies the target node with the "first node for directory look-up" -- i.e., a node of the (primary) virtual network for virtual directory look-up. However, this seems to be at odds with the text of paragraph a, which indicates that the target node acts to find a number of documents (as opposed to virtual directories) matching the query.

According to claim 1, item look-up is performed by the nodes of the second-recited virtual network -- not the nodes of the first-recited virtual network which perform virtual directory look-up.

Paragraph b describes forwarding the query if not enough documents are found at the first node, and paragraph c describes returning the results. Nothing in the description of the target node is relevant to a node of the primary virtual network for virtual directory look-up.

The Final Office Action also identifies target node steps a-c with "wherein the second node is for item look-up" (actual claim wording: "a node of a virtual network for item look-up"). However, this allegation appears to conflict with the Final Office Action's concurrent allegation that these same paragraphs describe a node of the primary virtual network for virtual directory look-up. Paragraphs a and b appear to describe item look-up. The Final Office Action points out that a node may be a logical entity. However, the target node is only described in relation to a single look-up function, and it is not understood how this single function can be mapped to the two very different functions of directory look-up and item look-up. The Examiner to-date still has not indicated whether the description of the target node is viewed as describing the activity of item look-up or directory look-up, despite Applicant's requests for clarification.

The Examiner also seeks to identify target node paragraph c with "in response to an enquiry message that identifies the virtual directory with which the node is associated, generating a reply message identifying a computer that the node for directory look-up is located on" (underscored part omitted by the Examiner's chopped quotation). But the

Examiner still has not explained how the wording of paragraph c, which contains no reference to "computer" or "computer identity," corresponds to generating a reply message identifying a computer. Of course, the results returned according to paragraph c in any event relate to the documents found in paragraphs a and b, not computer identifiers.

The Final Office Action's allegations regarding target node paragraph c is a good illustration of how the "shopping list" approach to seeking out and retrieving extracted bits and pieces of Applicant's claim language fails. In seeking to break the claim down into phrases that represent less than a single feature, the Final Office Action deviates from the task of judging the merits of the invention as an integral set of related features towards a process of finding random phrases in unrelated passages spread among several disparate documents. As the MPEP and related case law (and 35 U.S.C. § 103 itself) make clear, the obviousness determination must pertain to a claim "as a whole." Not only does the Examiner's approach of breaking down the claims into individual limitations and then further breaking down those individual limitations into fragments of a thought rely exclusively on improper hindsight, but is also tends to result in the "finding" of the extracted bits of features in the strangest -- and most non-combinable -- of places.

From section 3.3 Requesting Node:

The Examiner seeks to identify requesting node paragraphs a-c with "in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network." It is noted that the

"network" referred to here is the network for directory look-up. This activity would logically seem to be more appropriate to the receiving node, as it describes the action of a node receiving a request that is intended for a node in a different directory. In any case, requesting node paragraphs a-c do not teach forwarding a request. If the requested node does not exist, the enquiry will fail -- but there is no teaching or suggestion of forwarding the failed enquiry.

The Examiner identifies requesting node paragraphs a-b with "a node of a virtual network for item look-up...comprising... linking data comprising addresses of [wherein] other such nodes for item look-up [are] each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory" (underscored parts omitted by the Examiner's chopped quotation/paraphrasing; parts in [] inserted by Examiner).

The requesting node is characterized at paragraphs *a* and *b* by mapping document keywords to semantic (i.e., document) categories and finding the clusters of nodes to which those semantic categories are mapped. No mention can be found in paragraphs *a* and *b* of linking data comprising the addresses of nodes.

The requesting node is further characterized at paragraph *c* by selecting a node at random. Arguably, selection at random does not require the linking data which is not, in any case, described with reference to the requesting node.

The Examiner has not identified any text in these paragraphs as teaching the linking data despite Applicant's requests. Perhaps the reason that the underlined features

were omitted from the paraphrasing of the above-identified claim language in the Final Office Action is that there is no teaching or suggestion of the same.

The Examiner also identifies requesting node paragraphs b-c with "send to a node of the virtual network for directory look-up an enquiry message identifying the directory" (emphasis added). Elsewhere, the Final Office Action equates the text of paragraph b with the activity of a node of the second virtual network for item look-up. Although the Final Office Action points out that a node may be a logical entity, the Examiner to-date has not explained whether the text of paragraph *b* -- which must relate to a single activity category -- relates to the activity of item look-up or directory look-up.

Kwon describes a hierarchical network of ordinary "leaf" nodes and supernodes (see Figure 1). The file search described in Kwon as the "3-step" method involves storing on supernodes information (directory record) relating to a file. The directory record is stored in a so-called "file directory" on each supernode.

In relation to "(b) in response to an enquiry message identifying the virtual directory with which the node is associated to generate a reply message identifying the computer," it is presumed that the Final Office Action is referring to the first section of 3.2, at the bottom of the first side, i.e., "[a] node first starts the publishing procedure by sending a message to the supernode to which it is directly connected. The message contains the following data record <... IP address>" (emphasis added).

Here, Kwon is not describing a process to locate a desired data item, but rather is referring to the publishing of the details of a file to allow it to be located more easily in a later search stage. Those skilled in the art would understand from this description in

Kwon that, in order to enable the finding of a file, one should publish details of the file to supernodes that do not hold the file. The details published according to Kwon include the address of the node that does hold the file.

Claim 1 simply does not involve such a publishing step. One of ordinary skill in the art at the time the invention was made would see no incentive to introduce a similar publishing step into Bonsma or Triantafillou, at least because they each function efficiently in their own respective modes without the need for supernodes and require no publishing procedure(s). Differently stated, one skilled in the art at the time of the invention would not incorporate Kwon's publishing techniques into Bonsma or Triantafillou to solve a problem that does not exist and/or is already solved in a fundamentally different way.

Of course, Kwon actually is incompatible with the operation of exemplary embodiments of claim 1 -- where sending a request for a virtual directory associated with a data item to a node that is associated with the virtual directory (and therefore stores one or more of the desired data items) results in the node that is associated with the virtual directory returning the identity of the computer on which the node is located. This allows the retrieval means of exemplary embodiments of claim 1 to simply send to the identified computer an enquiry message identifying the data item and to receive the desired item in reply. Kwon does not teach features that correspond to the above-quoted features of claim 1 and disadvantageously involves an extra overhead (e.g., in setting up and maintaining additional, so-called supernodes as a central look-up for file locations) not incurred by exemplary embodiments of claim 1.

The Final Office Action concedes that Bonsma does not teach or suggest that "at least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to" take the particular actions called for in claim 1. Nor are corresponding features found in the other relied-upon references, alone or when taken in combination, because the fundamental specific hierarchy of the virtual network for directory look-up and the virtual network for directory look-up are not taught or suggested in any of the cited art. Applicant further elaborates on this point, below.

Claim 1 also recites "software operable (a) in response to an enquiry message that identifies another of the items {i.e., not an item "with which the node is associated"}, forwarding the message to another node for item look-up of the network" (text added in {} to put this part of claim 1 in context). The phrase "another of the items" stands in contrast to "the item with which the node for the look-up item is associated" which occurs in the immediately following part (b) of this part of the claim. The phrase "another of the items" clearly defines an item that is not an item "with which the node for item look-up is associated."

The supernodes of Kwon do not function in this way, e.g., they do not decide whether a request is for data that they hold, and they do not respond by either providing the data item or forwarding the request to another node. The Final Office Action relies on Kwon's "step-by-step" method and characterizes this method as "forwarding if local look-up fails." But what is described in Kwon is not local look-up (i.e., checking to see if the node itself has the data item requested), but rather what is described is remote look-up (i.e., look-up carried out by a remote supernode).

Claim 1 does not have any supernodes that store information about items with which other nodes are associated. The nodes of the distributed computer system in exemplary embodiments of claim 1 respond to enquiries according to information relating to their own status -- i.e., whether they have the sought-after data item.

With respect to the claim 1 requirement for "(b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the identified item," the Final Office Action now refers to the "3-step" method of Kwon. The Final Office Action essentially argues that it would have been obvious for one of ordinary skill in the art at the time the invention was made to combine these two disparate teachings (i.e., the "step-by-step" method and the "3-step" method) from the same document to arrive at features of the claimed invention.

This argument fails for at least the simple reason that it did not occur to the authors of Kwon -- not just ones of ordinary skill in the art, but workers who were leading advances in the field -- to combine these aspects that Kwon devised as parts of separate and distinct approaches. It cannot, therefore, be maintained that one of ordinary skill in the art would be motivated to combine these teachings.

As the Final Office Action now recognizes, even the proposed patchworked combination of Bonsma/Triantafillou/Kwon does not teach:

> "wherein a reply message includes the item;
>
> and wherein at least one computer has retrieval means to:
>
> ii) upon receipt of a reply message thereto, to send to a computer an enquiry message identifying the item; and

iii) to receive the reply message containing the item."

As with earlier parts of the Final Office Action, this paraphrasing does not exactly track the actual claim language of claim 1. It appears that the Examiner has conveniently changed the wording to suit the assertions being made in the Final Office Action.

In any event, Adar does not supply the previously noted deficiencies of the first three references. Furthermore, Adar's teachings at page 5 with respect to "get/push messages" does not appear to have anything to do with receiving a query identifying a directory and an item within that directory -- nor does it seem to teach anything like responsively sending to a node of the virtual network for directory look-up an inquiry message identifying the directory -- much less the latter two recitations of claim 1 that require sending a reply message back to the computer identified in the reply message, an enquiry message identifying the item and, subsequently to receive the reply message containing the item.

In short, the Examiner's extraction of the "teachings" of this fourth "combined" reference is no better than that of the first three. And, once again, even if the hypothesized "combination" of selected bits and pieces from all four of these references is somehow "combined" into an operable system of some undefined sort, it would still fail to teach or suggest the combination of elements set forth in Applicant's claim 1.

The need to rely on fragments taken from four different documents in order to locate text that can be separately compared with each component phrase of claim 1 indicates the novel and non-obvious nature of the claimed invention. None of the cited documents describes or even suggests the system of claim 1. The various citations --

even when combined in the complex relationship and often internally inconsistent manner proposed in the Final Office Action -- do not provide the specific combination of features devised by the present inventors and do not provide the special benefits provided by the claimed invention. Moreover, the proposed combination of teachings lacks motivation. Without Applicant's claim wording as a guide (e.g., as input to post hoc word searching algorithms then applied to thousands of potential targets), there is insufficient incentive for one of only ordinary skill in the art at the time of the invention to combine the many unrelated parts of the many unrelated references selected in the precise way that is proposed in the Final Office Action -- even though that proposed combination inevitably also fails to approximate Applicant's claim 1.

The above deficiencies noted with respect to claim 1 are also present with respect to independent claims 2, 6, 7, 11, and 13. Applicant therefore requests that this four-way § 103 rejection be reversed as to all independent claims (and their respective dependents).

B. Claims 5, 9-10, and 13-15 Each Are Not "Obvious" Over Bonsma, Triantafillou, Kwon, Adar, and Christenson.

Claims 5, 9-10, and 13-15 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, and Christenson et al. (U.S. Publication No. 2002/0112008). This rejection is erroneous and should be reversed for at least the following reasons.

Fundamental deficiencies with respect to the four-way combination of Bonsma, Triantafillou, Kwon, and Adar have described in detail above. The further introduction of Christenson, even if appropriate (which Applicant certainly does not concede), still

would fail to make up for these fundamental deficiencies. Thus, Applicant respectfully requests that this five-way § 103 rejection be reversed.

C. Claims 19-20 Each Are Not "Obvious" Over Bonsma, Triantafillou, Kwon, Adar, and Bonsma '669.

Claims 19-20 stand rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, and Bonsma '669 (WO 03/034669). This rejection is erroneous and should be reversed for at least the following reasons.

Fundamental deficiencies with respect to the four-way combination of Bonsma, Triantafillou, Kwon, and Adar have described in detail above. The further introduction of Bonsma '669, even if appropriate (which Applicant certainly does not concede), still would fail to make up for these fundamental deficiencies. Thus, Applicant respectfully requests that this five-way § 103 rejection be reversed.

D. Claim 21 Is Not "Obvious" Over Bonsma, Triantafillou, Kwon, Adar, Bonsma '669 and Yemini.

Claims 21 stands rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Bonsma, Triantafillou, Kwon, Adar, Bonsma '669, and Yemini et al. (U.S. Publication No. 2002/0163889). This rejection is erroneous and should be reversed for at least the following reasons.

Fundamental deficiencies with respect to the four-way combination of Bonsma, Triantafillou, Kwon, and Adar have described in detail above. The further introduction of Bonsma '669 and/or Yemini, even if appropriate (which Applicant certainly does not

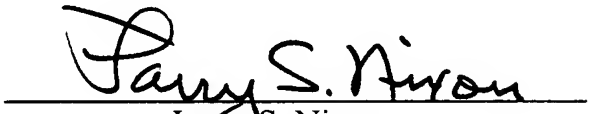
concede), still would fail to make up for these fundamental deficiencies. Thus, Applicant respectfully requests that this six-way § 103 rejection be reversed.

CONCLUSION

In conclusion it is believed that the rejections of claims 1-21 are erroneous and should be reversed.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: 

Larry S. Nixon
Reg. No. 25,640

LSN:jr
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100

(VIII) CLAIMS APPENDIX

1. A distributed computer system comprising a plurality of computers, said system comprising:

a plurality of computers, each storing data items, each data item being assigned to one of a plurality of virtual directories;

each computer that has a said data item stored thereon having at least one node of a virtual network for directory look-up, said node for directory look-up comprising

(i) data identifying that one of the plurality of virtual directories with which the node for directory look-up is associated,

(ii) linking data comprising addresses of other such nodes for directory look-up, and

(iii) software operable,

(a) in response to an enquiry message that identifies another of the virtual directories forwarding the message to another node for directory look-up of the network, and

(b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on:

each computer that has a said data item stored thereon having, for each item stored thereon, a node of a virtual network for item look-up, said node for item look-up comprising

(i) data identifying the item with which the node for item look-up is associated,

(ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network, and

(b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the identified item; and

wherein at least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to

(i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory,

(ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item, and

(iii) to receive the reply message containing the item.

2. A distributed computer system comprising a plurality of computers, said system comprising:

a plurality of computers, each storing data items, each data item being assigned to one of a plurality of virtual directories;

each computer that has a said data item stored thereon having at least one node of a virtual network for directory look-up, said node for directory look-up comprising

(i) data identifying that one of a plurality of the virtual directories with which the node for directory look-up is associated,

(ii) linking data comprising addresses of other such nodes for directory look-up, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network, and

(b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on;

each computer that has a said data item stored thereon having, for each item stored thereon, a node of a virtual network for item look-up, said node for directory look-up comprising

(i) data identifying the item with which the node for directory look-up is associated,

(ii) linking data comprising addresses of other such nodes for directory look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the items forwarding the message to another node for directory look-up of the network,

(b) in response to an enquiry message that identifies the item with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for item look-up is located on, and

(c) in response to a request message that identifies the item with which the node for directory look-up is associated, generating a reply message including the item; and

wherein at least one computer has retrieval means responsive to receipt of a query identifying a directory and an item within that directory to

(i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory,

(ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item, and

(iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item.

3. A computer system according to claim 1 in which each computer having retrieval means includes also secondary retrieval means operable

(a) upon receipt of a reply message identifying a computer having one or more items in a particular directory to identify further computers having one or more items in that directory; and

(b) to create a list of items in that directory.

4. A computer system according to claim 3, wherein each computer that has a said data item stored thereon also has at least one node of a secondary virtual network for directory look-up, such that said nodes of a secondary virtual network for directly look-up together form a respective secondary virtual network for each virtual directory, wherein said node of a secondary virtual network for directly look-up comprising a data storage area for containing a list of addresses of other nodes of the secondary virtual network that have items in the same directory and said node of a secondary virtual network for directly look-up is responsive to enquiry messages to return a message containing the addresses of the list; and

wherein the secondary retrieval means is operable, for identifying further computers having one or more items in the directory in question, to send an enquiry message to the node identified by the reply message and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message.

5. A computer system according to claim 1, wherein some of said directories are assigned, as subdirectories, to another of said directories and wherein each computer having retrieval means also includes:

(a) first subdirectory retrieval means responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory; and

(b) second subdirectory retrieval means connected to receive an address identified by the first subdirectory retrieval means and operable in response thereto to identify further computing nodes having items in at least one subdirectory assigned to the same directory.

6. A computer for use in a distributed computer system comprising a plurality of computers, said computer comprising:

a data store which stores data items, each data item being assigned to one of a plurality of virtual directories;

at least one node of a virtual network for directory look-up, said node for directory look-up comprising

(i) data identifying that one of the plurality of virtual directories with which the node for directory look-up is associated,

(ii) linking data comprising addresses of other such nodes for directory look-up, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network, and

(b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying a computer that the node for directory look-up is located on;

for each item stored on said computer, a node of a virtual network for item look-up, said node for item look-up comprising

(i) data identifying the item with which the node for item look-up is associated,

(ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network, and

(b) in response to an enquiry message that identifies the item with which the node for item look-up is associated, generating a reply message including the item; and

retrieval means responsive to receipt of a query identifying a directory and an item within that directory to

(i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory,

(ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item, and

(iii) to receive the reply message containing the item.

7. A computer for use in a distributed computer system comprising a plurality of computers, said computer comprising:

a data store which stores data items, each data item being assigned to one of a plurality of virtual directories;

at least one node of a virtual network for directory look-up, said node for directory look-up comprising

(i) data identifying that one of a plurality of the virtual directories with which the node for directory look-up is associated,

(ii) linking data comprising addresses of other such nodes for directory look-up, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the virtual directories, forwarding the message to another node for directory look-up of the network, and

(b) in response to an enquiry message that identifies the virtual directory with which the node for directory look-up is associated, generating a reply message identifying the computer that the node for directory look-up is located on;

for each item stored on said computer, a node of a virtual network for item look-up, said node for item look-up comprising

(i) data identifying the item with which the for item look-up node is associated,

(ii) linking data comprising addresses of other such nodes for item look-up each associated with an item assigned to the same virtual directory, whereby said linking data together define a plurality of virtual networks for item look-up, each of which networks corresponds to a respective different virtual directory, and

(iii) software operable

(a) in response to an enquiry message that identifies another of the items, forwarding the message to another node for item look-up of the network,

(b) in response to an enquiry message that identifies the item with which the node for item look-up is associated generating a reply message identifying a computer that the node for item look-up is located on, and

(c) in response to a request message identifying the item with which the node for item look-up is associated, generating a reply message including the item; and

retrieval means responsive to receipt of a query identifying a directory and an item within that directory to

(i) send to a node of the virtual network for directory look-up an enquiry message identifying the directory;

(ii) upon receipt of a reply message thereto, to send to the computer identified in the reply message an enquiry message identifying the item;

(iii) upon receipt of a reply message thereto, to send to the computer identified in the reply message a message requesting the item.

8. A computer according to claim 6 including also secondary retrieval means operable:

(a) upon receipt of a reply message identifying a computer having one or more items in a particular directory to identify further computers having one or more items in that directory; and

(b) to create a list of items in that directory.

9. A computer according to claim 6, wherein some of said directories are assigned, as subdirectories, to another of said directories and wherein the computer also includes:

(i) first subdirectory retrieval means responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory; and

(ii) second subdirectory retrieval means connected to receive an address identified by the first subdirectory retrieval means and operable in response thereto to identify

further computing nodes having items in at least one subdirectory assigned to the same directory.

10. A computer according to claim 9 in which the retrieval means is operable to compile a composite list of said subdirectories.

11. A distributed computer system comprising:

a plurality of computing nodes, wherein each computer stores data items, each data item being assigned to one of a plurality of virtual directories;

first retrieval means responsive to input of a directory name to identify a computing node having items in that directory;

second retrieval means connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in the same directory;

wherein each computing node having items in a given directory has associated with it a data storage area for containing addresses for other computing nodes having items in the same directory and is responsive to enquiry messages to return a message containing the addresses of the list; and

wherein the second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or

in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in the directory in question.

12. A distributed computer system according to claim 11 in which the retrieval means is operable to retrieve from each of said identified plurality of computing nodes a list of items stored thereon, and to compile a composite list of said items.

13. A distributed computer system comprising:

a plurality of computing nodes, wherein each computer stores data items, each data item being assigned to one of a plurality of virtual directories, some of said directories being assigned, as subdirectories, to another of said directories;

first retrieval means responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory;

second retrieval means connected to receive an address identified by the first retrieval means and operable in response thereto to identify further computing nodes having items in at least one subdirectory assigned to the same directory;

wherein each computing node having items in at least one subdirectory assigned to a given directory has associated with it a data storage area for containing addresses for other computing nodes having items in at least one subdirectory assigned to the same directory and is responsive to enquiry messages to return a message containing the addresses of the list; and

wherein the second retrieval means is operable to send an enquiry message to the node identified by the first retrieval means and upon receipt of a response to iteratively send enquiry messages to addresses contained in the response to that enquiry message or in a response to a subsequent enquiry message, thereby identifying a plurality of computing nodes having items in subdirectories of the directory in question.

14. A distributed computer system according to claim 13 in which the retrieval means is operable to compile a composite list of said subdirectories.

15. A distributed computer system according to claim 11 wherein some of said directories are assigned, as subdirectories, to another of said directories and wherein each computer having retrieval means also includes:

first subdirectory retrieval means responsive to input of a directory name to identify a computing node having items in at least one subdirectory assigned to that directory; and

second subdirectory retrieval means connected to receive an address identified by the first subdirectory retrieval means and operable in response thereto to identify further computing nodes having items in at least one subdirectory assigned to the same directory.

16. A distributed computer system according to claim 11 wherein:

the first retrieval means is formed by a primary network of virtual nodes, each node being defined by a list of links to other nodes of the secondary network, each entry in the list including a label and address of the respective other node; and

each node includes (i) means responsive to receipt of a request message containing a label to propagate the request message within the network, and (ii) means responsive to receipt of a request message containing a label matching the label of the node receiving it to generate a reply message.

17. A distributed computer system according to claim 11 in which the second retrieval means is formed by a secondary network of virtual nodes, each node being defined by a list of links to other nodes of the primary network, each entry in the list including an address of the respective other node; and wherein each node includes means responsive to receipt of a request message to generate a reply message containing the addresses of the list.

18. A distributed computer system according to claim 16

in which the second retrieval means is formed by a secondary network of virtual nodes, each node being defined by a list of links to other nodes of the primary network, each entry in the list including an address of the respective other node; and wherein each node includes means responsive to receipt of a request message to generate a reply message containing the addresses of the list; and

in which the reply message generated by a node of the primary network includes the address of that node of the secondary network which is associated with the node generating the reply message.

19. A distributed computer system according to claim 16, wherein

the second retrieval means is formed by a secondary network of virtual nodes, each node being defined by a list of links to other nodes of the primary network, each entry in the list including an address of the respective other node; and wherein each node includes means responsive to receipt of a request message to generate a reply message containing the addresses of the list;

each node of the primary network includes means operable to initiate and to propagate exploratory messages each containing the label and address of the initiating node of the primary network; and

each node is operable upon receipt of an exploratory message containing a label matching that of the receiving node and an address not matching that of the receiving node to generate a notification message for addition of a link to the secondary network, said notification message identifying the node initiating the exploratory message and containing the address of the node of the secondary network associated with the receiving node.

20. A distributed computer system according to claim 19, in which the notification message contains, as destination, the address of the initiating node, and the

initiating node is operable upon receipt thereof to forward to the node of the secondary network associated with the initiating node a message requesting addition of a link between it and the node having the address contained in the notification message.

21. A distributed computer system according to claim 17 in which each node of the secondary network includes processing means programmed to perform the following operations:

receiving messages;

responding to messages requesting information about the contents of the list;

complying with received requests to remove an address from the list and insertion of another address into the list; and

in response to receipt of a message requesting a link between the node and a second node:

(A) generating a message to the second node requesting information about the contents of its list;

(B) determining whether both the first node and second node has in each case a number of addresses in its list which is less than the predetermined number;

(C) in the event that this condition is satisfied, inserting into its list the address of the second node and generating a message to the second node requesting the second node to add to its list the address of the node;

(D) in the event that this condition is not satisfied, determining whether the node has a number of addresses in its list which is at least two less than the predetermined number, and if so

(a) selecting from the list of the second node the address of a third node;

(b) inserting the address of the second node into the list of the first node and inserting the address of the third node into the list of the first node;

(c) generating a message to the second node requesting the removal of the address of the third node from the list of the second node and insertion of the address of the node; and

(d) generating a message to the third node requesting the removal of the address of the second node from the list of the third node and insertion of the address of the node.

(IX) EVIDENCE APPENDIX

None.

(X) RELATED PROCEEDINGS APPENDIX

None.